

CLAIMS

1. (Previously Presented) A method of apportioning additional thread activation time to at least one application comprising at least two computer application threads that experience activation time budget overruns in a system where the at least one application monitors and controls functions, the method comprising:

setting a variable to a system pad time value predetermined before execution of the application; and

during the execution of the at least two computer application threads of the at least one application,

activating a first application thread for a first actual activation time,

comparing the first actual activation time with a first activation time budget predetermined before execution of the application,

if the first actual activation time exceeds the first activation time budget, adjusting the system pad time value to thereby obtain an updated pad time value,

activating a second application thread for a second actual activation time,

comparing the second actual activation time with a second activation time budget predetermined before execution of the application, and

if the second actual activation time exceeds the second activation time budget, adjusting the system pad time value to thereby obtain the updated pad time value.

2. (Previously Presented) The method of Claim 1, further comprising:
setting the variable to the updated pad time value.

3. (Previously Presented) The method of Claim 2, further comprising:
initiating a predetermined response if the updated pad time value is adjusted to a predetermined response initiation value.

4. (Previously Presented) The method of Claim 3, wherein the predetermined response initiation value is a value less than zero.

5. (Previously Presented) The method of Claim 1, further comprising:
resetting the variable to the predetermined system pad time value at a predetermined periodicity.

6. (Previously Presented) The method of Claim 5, wherein the predetermined periodicity is each time the application is executed.

7. (Currently Amended) The method of Claim 1, wherein the first ~~thread~~ activation time budget comprises:

a first thread execution time; and

a first thread pad time.

8. (Previously Presented) The method of Claim 7, further comprising:
subtracting the first thread pad time from the first activation time budget to obtain a first initial thread timer value;

initializing a first thread timer to the first initial thread timer value; and

enabling the first thread timer upon activation of the first application thread.

9. (Previously Presented) The method of Claim 8, further comprising:

disabling the first thread timer upon completion of the first actual activation time to thereby obtain a first final thread timer value; and

adding the first thread pad time to the first final thread timer value to determine a first activation time difference value between the first actual activation time and the first activation time budget.

10. (Previously Presented) The method of Claim 9, wherein the first thread timer is operable to decrement from the first initial thread timer value to the first final thread timer value.

11. (Previously Presented) The method of Claim 9, further comprising:

determining whether the first activation time difference value is positive or negative; and
if the first activation time difference is negative, adding the first activation time difference value to the system pad time value to obtain an updated pad time value.

12. (Currently Amended) The method of Claim 11, further comprising:

determining whether the updated pad time value is positive or negative; and
if the updated pad time value is [[a]] negative, initiating a predetermined response.

13. (Currently Amended) A system for apportioning additional thread activation time to at least two application threads of an application that experiences activation time budget overruns while executing on a computer, comprising:

a memory adapted to store at least a system pad time value therein, wherein the system pad time value is predetermined before execution of the application; and

a processor in operable communication with the memory and operable to:

(i) activate a first application thread for a first actual activation time,

(ii) compare the first actual activation time of the first application thread with a first activation time budget, wherein the first activation time budget is predetermined before execution of the application,

(iii) if the first actual thread activation time exceeds the first ~~thread~~ activation time budget, adjust the system pad time value to obtain an updated pad time value,

(iv) activate a second application thread for a second actual activation time,

(v) compare the second actual activation time of the second application thread with a second activation time budget, wherein the second activation time budget is predetermined before execution of the application, and

(vi) if the second actual activation time exceeds the second activation time budget, adjust the system pad time value to obtain the updated pad time value.

14. (Previously Presented) The system of Claim 13, wherein the processor is further operable to store the updated pad time value in the memory.

15. (Previously Presented) The system of Claim 14, wherein the processor is further operable to initiate a predetermined response if the updated pad time value is adjusted to a predetermined response initiation value.

16. (Previously Presented) The system of Claim 15, wherein the predetermined response initiation value is a value less than zero.

17. (Previously Presented) The system of Claim 13, wherein the processor is further operable to reset the variable to the predetermined system pad time value at a predetermined periodicity.

18. (Previously Presented) The method of Claim 17, wherein the predetermined periodicity is each time the application is executed.

19. (Currently Amended) The system of Claim 13, wherein the first ~~thread~~ activation time budget comprises:

a first thread execution time; and

a first thread pad time.

20. (Currently Amended) The system of Claim 19, further comprising:

a first thread timer in communication with the processor and operable, in response to an enable signal from the processor, to determine the first actual activation time, wherein the processor is further operable to:

subtract the first thread pad time from the first ~~thread~~ activation time budget to obtain a first initial thread timer value,

initialize the first thread timer to the first initial thread timer value, and

supply the enable signal to the first thread timer upon activation of the first application thread.

21. (Previously Presented) The system of Claim 20, wherein the processor is further operable to:

supply a disable signal to the first thread timer upon completion of the first actual activation time to thereby stop the first thread timer and obtain a first final thread timer value; and

add the first thread pad time to the first final thread timer value to determine a first activation time difference value between the first actual activation time and the first activation time budget.

22. (Previously Presented) The system of Claim 18, wherein the first thread timer is operable to decrement from the first initial thread timer value to the first final thread timer value.

23. (Previously Presented) The system of Claim 18, wherein the processor is further operable to:

determine whether the first activation time difference value is positive or negative; and

if the first activation time difference is negative, add the first activation time difference value to the system pad time value to obtain an updated pad time value.

24. (Previously Presented) The system of Claim 23, wherein the processor is further operable to:

determine whether the updated pad time value is positive or negative; and
if the updated pad time value is negative, initiate a predetermined response.

25. (Currently Amended) A computer readable medium containing computer executable code for instructing a computer that executes at least one application comprising at least two computer application threads, which is configured to apportion additional thread activation time to the at least two computer application threads that experience activation time budget overruns, to perform the steps of:

setting a variable to a system pad time value, wherein the system pad time value is predetermined before execution of the at least one application;

activating a first application thread for a first actual activation time;

comparing the first actual activation time with a first activation time budget, wherein the first activation time budget is predetermined before execution of the at least one application;

if the first actual activation time exceeds the first activation time budget, adjusting the system pad time value to thereby obtain an updated pad time value;

activating a second application thread for a second actual activation time;

comparing the second actual activation time with a second activation time budget, wherein the second activation time budget is predetermined before execution of the at least one application; and

if the second actual activation time exceeds the second activation time budget, adjusting the system pad time value to thereby obtain the updated pad time value[[,]].

26. (Previously Presented) The computer readable medium of Claim 25, containing computer executable code for instructing a computer to perform the further steps of:

setting the variable to the updated pad time value.

27. (Previously Presented) The computer readable medium of Claim 26, containing computer executable code for instructing a computer to perform the further steps of:

initiating a predetermined response if the updated pad time value is adjusted to a predetermined response initiation value.

28. (Previously Presented) The computer readable medium of Claim 27, wherein the predetermined response initiation value is a value less than zero.

29. (Previously Presented) The computer readable medium of Claim 25, containing computer executable code for instructing a computer to perform the further steps of:

resetting the variable to the predetermined system pad time value at a predetermined periodicity.

30. (Previously Presented) The computer readable medium of Claim 29, wherein the predetermined periodicity is each time the application is executed.

31. (Previously Presented) The computer readable medium of Claim 25, wherein the first activation time budget comprises:

a first thread execution time; and

a first thread pad time.

32. (Previously Presented) The computer readable medium of Claim 31, containing computer executable code for instructing a computer to perform the further steps of:

subtracting the first thread pad time from the first activation time budget to obtain a first initial thread timer value;

initializing a first thread timer to the first initial thread timer value; and

enabling the first thread timer upon activation of the first application thread.

33. (Previously Presented) The computer readable medium of Claim 32, containing computer executable code for instructing a computer to perform the further steps of:

disabling the first thread timer upon completion of the first actual activation time to thereby obtain a first final thread timer value; and

adding the first thread pad time to the first final thread timer value to determine a first activation time difference value between the first actual activation time and the first activation time budget.

34. (Previously Presented) The computer readable medium of Claim 33, wherein the first thread timer is operable to decrement from the first initial thread timer value to the first final thread timer value.

35. (Previously Presented) The computer readable medium of Claim 33, containing computer executable code for instructing a computer to perform the further steps of:

determining whether the first activation time difference value is positive or negative; and
if the first activation time difference value is negative, adding the first activation time difference value to the system pad time value to obtain an updated pad time value.

36. (Currently Amended) The computer readable medium of Claim 35, containing computer executable code for instructing a computer to perform the further steps of:

determining whether the updated pad time value is positive or negative; and
if the updated pad time value is ~~[[a]]~~ negative, initiating a predetermined response.

37. (Currently Amended) ~~[[A]] An aircraft avionics device, system for apportioning additional thread activation time to application threads of a system that controls and monitors functions, the system comprising at least two computer application threads of at least one application that experiences activation time budget overruns while executing on a computer, the at least one application comprising:~~

a central processing unit (CPU);

memory;

at least one application that experiences activation time budget overruns while executing on the CPU and using the memory, the at least one application comprising:

at least two computer application threads;

setting means for setting a variable to a predetermined system pad time value;

thread activation means for activating a first computer application thread of the at least one application for a first actual activation time and for activating a second computer application thread of the at least one application for a second actual activation time;

activation time determination means, in operable communication with the thread activation means, for determining the first actual application time of the first application thread and for determining the second actual application time of the second application thread;

comparison means, in operable communication with the activation time determination means, for comparing the first actual application time of the first application thread with a first activation time budget and for comparing the second actual application time of the second application thread with a second activation time budget; and

adjustment means, in operable communication with the comparison means, for adjusting the system pad time value to thereby obtain an updated pad time value if either the first actual thread activation time exceeds the first ~~thread~~ activation time budget or the second actual thread activation time exceeds the second ~~thread~~ activation time budget.